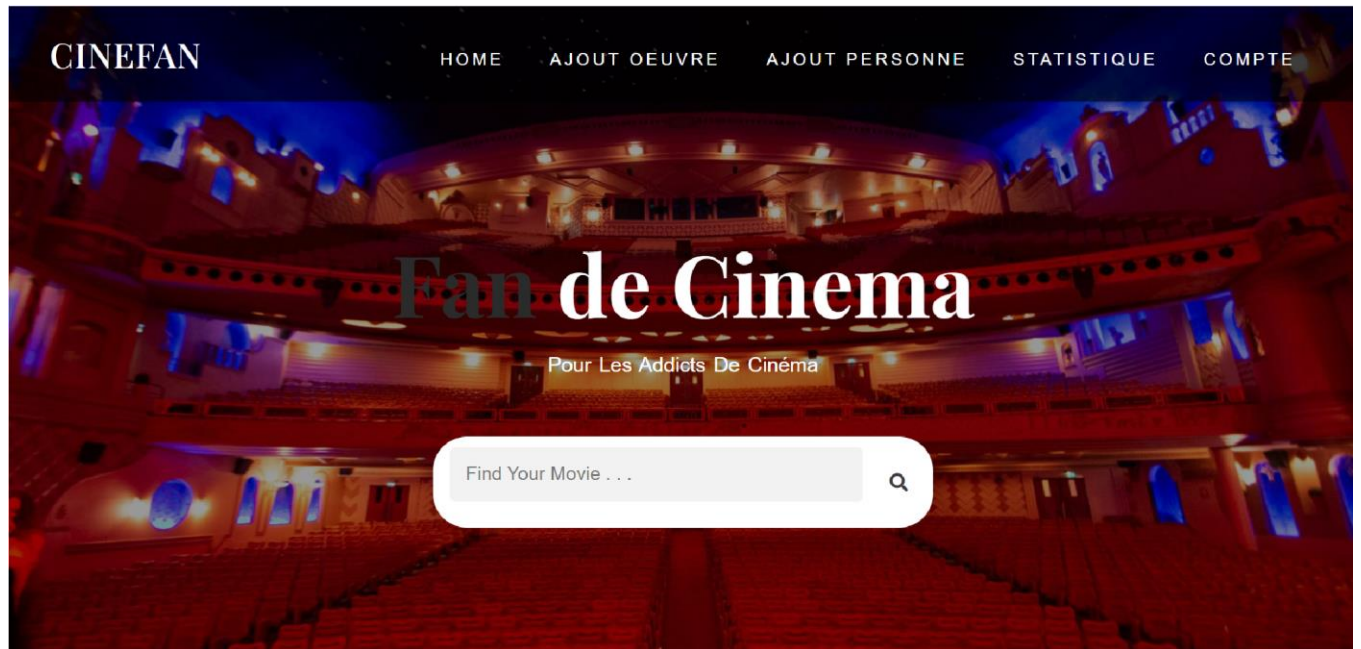


RAPPORT CINEFAN

Projet de Base de données



Mickaël RAKOTOARISON

Iandrianina RAKOTOMALALA

19.12.2021

L2 INFORMATIQUE

Table de matières

Table des matières

Mickaël RAKOTOARISON	0
INTRODUCTION	2
MANUEL UTILISATEUR	5
DESCRIPTION TECHNIQUE	5
MODIFICATIONS DU DUMP	6
ORGANISATION DU TRAVAIL	6
CHOIX DANS LA CONCEPTION	7
LES PARTIE NON TRAITE	7

INTRODUCTION

Le projet consiste à créer un site web pour les fans de cinéma qui sera relié à une base de données. Un internaute peut consulter le site en étant inscrit ou pas, et un utilisateur inscrit aura ainsi un identifiant et un mot de passe afin de se connecter. Un utilisateur pourra ajouter des fiches concernant une œuvre (film/série) ou un acteur/réalisateur. Ces fiches ajoutées seront retranscrites dans la base de données ainsi que les informations sur l'utilisateur.

ETAPE 1 : Modélisation

❖ Premier modèle :

➤ Utilisateur :

Cette entité contient toutes les informations relatives à un utilisateur (pseudo, mot de passe, **ses favoris**)

➤ Fiche:

Entité qui a pour attribut un numéro fiche comme PK. Cette entité sera associée à l'utilisateur qui l'a postée, mais également à une œuvre (film ou série) et à une personne (acteur ou réalisateur).

➤ Personne:

Cette entité a pour attributs un identifiant comme PK, ainsi que son nom et son prénom.

Ensuite, les entités acteur et réalisateur hériteront d'elle, et contiendront également respectivement un identifiant idacteur ainsi que son rôle et un idrealisateur.

➤ Oeuvre:

Les attributs de cette entité sont : un identifiant qui sera son PK, ainsi qu'un titre et sa date de sortie. Elle aura également deux entités qui hériteront d'elle qui sont film et

série. Ces deux derniers seront reliés par une association “est un” permettant de savoir si l’un des deux donne suite à une autre.

➤ Genre

Cette entité contient un genre et est associée à une œuvre.

Nous remarquons que le premier modèle est très bancal et incomplet. Les attributs sont manquants et il y a des entités/associations inutiles telles que l’entité *fiche*, l’association *est un* entre film et série.

❖ Schéma final:

➤ Acteur/réalisateur:

Représente soit un acteur soit un réalisateur ou même les deux et a pour attributs : un identifiant en PK, son nom, son prénom, sa nationalité

➤ Galerie:

A pour PK son identifiant *idphoto*. Elle sera associée à un acteur ou à un film/série.

➤ Saison/Volet:

A pour attribut le numéro de saison/volet et est associé à acteur/réalisateur présent(s) dans le film ainsi que le film/série qu’il représente ainsi que la date de sortie, la durée et un synopsis.

➤ Commente

➤ Utilisateur:

Représente les données d’un utilisateur : pseudo, mot de passe, email, le type d’utilisateur (admin ou non).

➤ Genre:

Contient le genre et est associé à un film/série.

ETAPE 2: Création de la base de données

Traduction du modèle en modèle relationnel

:

Entités :

ActeurRealisateur(id_personne, nom, prenom, naissance, nationalité)

SaisonVolet(id_SaisonVolet, #id_oeuvre, titre, date_sortie)

FilmSerie(id_oeuvre, titre, date_sortie, duree, descriptif)

Utilisateur(id_utilisateur, mdp, modérateur)

Genre(id_genre, #id_oeuvre, #id_personne)

Galerie(id_photo, url)

Associations : joue((id_personne, id_oeuvre, id_saisonvolet), role) pk : (id_personne,

id_oeuvre) realise((id_personne, id_oeuvre)) pk : (id_personne, id_oeuvre)

oeuvreAppartientGenre((id_oeuvre, id_genre)) commente((id_saisonvolet,

id_utilisateur, id_oeuvre), commentaire, note) pk : (id_saisonvolet, id_utilisateur,

id_oeuvre) suite((id_oeuvre_premier, id_oeuvre,suite) favoris((id_utilisateur,

id_oeuvre))

Choix (spécifiques) de représentation des données:

Pour les identifiants en PK tels que id_oeuvre, id_personne,etc nous avons choisis le type serial afin que lors de l'ajout de nouvelles valeurs, l'identifiant soit incrémenté directement sans avoir à le faire manuellement. Cependant lorsque ces identifiants sont référencés en clé étrangères dans d'autres tables, on utilise le type int.

Adaptations (vues pour les contraintes supplémentaires) :

CREATE VIEW mostpopular AS

(SELECT id_oeuvre, titre, date_sortie, descriptif, sum(note) AS notes

FROM filmserie NATURAL JOIN commente

GROUP BY id_oeuvre

ORDER BY notes DESC LIMIT 10

);

MANUEL UTILISATEUR

Un navigateur peut accéder au site via le lien :

perso-etudiant.u-pem.fr/~mickael.rakotoarison/ ou perso-etudiant.u-pem.fr/~rakotomalala/

Une fois sur la page d'accueil, un utilisateur peut s'inscrire puis naviguer en se connectant sur son compte.

Il peut alors :

- ajouter une fiche sur un film, mais le réalisateur étant requis, il doit d'abord faire une fiche sur ce dernier qui contiendra son nom, prénom, nationalité, date de naissance, etc.
- également ajouter des images liées aux saisons/volets d'un film/série. Il peut se déconnecter à la fin de sa session.
- Consulter les œuvres déjà inscrites sur le site.
- Consulter les statistiques

DESCRIPTION TECHNIQUE

- Cliquer sur une des barres de navigation redirige l'utilisateur vers une nouvelle page, contenue dans un fichier .php différent.

AJOUTS	PHP
Ajouter une nouvelle oeuvre	Un fichier .php sera exécuté derrière afin de rajouter les informations dans la base de données.
Ajouter une nouvelle personne (acteur/réalisateur)	Un fichier .php sera exécuté derrière afin de rajouter les informations dans la base de données.
Inscription	Lorsqu'un utilisateur s'inscrit, ses données sont inscrites dans la base de données.
Connexion	Lorsqu'un navigateur tente de se connecter, les informations qu'il saisit seront vérifiées et si elles sont correctes (telles qu'elles sont dans la base de donnée) alors il pourra commencer une session.

1. Nous avons créé un fichiers fonctions qui permette de proposer des œuvres dans la base de données et des acteur.etc.
2. Nous avons créé un fichier fonction qui permette de stoker une partit des variables de la base de données, elle est stoker sous forme de tableau associatif.
3. Pour l’affichage des images nous les stockons sur le serveur puis on y accède avec les liens stoker dans la table galerie, grâce à des requêtes sql, nous pouvons donc les récupérer.
4. Pour les ajouts d’œuvres il est primordial de définir d’abord les individus dans la base de données pour pouvoir les utiliser dans l’ajout de film.

MODIFICATIONS DU DUMP

1. La base de données ne contient plus d’informations c’est-à-dire que les tables sont initialement vides et que ce sont des structures vides.
2. La structure film/série contient un attribut pour le type afin de distinguer si c’est un film ou une série.
3. Dans la table Film/série : lorsqu’on ajoute un film/série, il sera directement ajouté dans la table *saisonvolet*. L'œuvre est alors définie dans la table filmserie par son premier volet/saison, les suites ne seront contenues que dans *saisonvolet*. Entre autres, le premier volet/saison est considéré comme la clé en général.
4. Dans la table Galerie : une photo sera ajoutée pour les saisons et volets et non pour l'œuvre en général.

ORGANISATION DU TRAVAIL

1. Modélisation : chacun a fait un schéma entité-association qui a été mis en commun avant d’être rendu.
2. Création de la base de donnée:
 - Mickael : dump + créations des tables
 - landrianina : modèle relationnel
3. Création du site:
 - Mickael : les bases du site + codes php + inscription et connexion à une session
 - landrianina : ajout oeuvres + ajout personne
4. Rapport : landrianina

CHOIX DANS LA CONCEPTION

1. Tout d'abord nous avons décidé que à chaque fois que l'on ajoute une œuvre dans la base de données, une saison/volet est créé correspondant à cette œuvre car on ne peut pas savoir à l'avance si c'est une suite de film ou une série à plusieurs saisons.
2. Le stockage des images dans le serveur.
3. L'ajout des personnes avant les œuvres.
4. Impossibilités de donner de nouveau genre.
5. L'ajout de plusieurs genre, acteur, réalisateur après la création du film dans une nouvelle page. La base de données est donc modifiée avec des ajouts.

LES PARTIES NON TRAITÉES

1. NOTE / COMMENTAIRE
2. BARRE DE RECHERCHE
3. STATISTIQUE
4. COMPTE MODERATION